

# Projet en informatique musicale — MUS3325X

## Musique Numérique

### Rodolphe Busuttil-Décousus

- Plan de travail :

**Début du projet :** Janvier 2017

**Fin du projet :**

- Mise à jour prévue : chaque semaine

**Rapport des mises à jours :**

- 1ère entrée : 23 janvier 2018 (mise en place de la mise en page du document, finalisation du Setting de Github)
- 2e entrée : 24 janvier 2018 (mise en place de l'énoncé et de la description.)

- Énoncé :

***Outil modulaire de composition et de performance vidéo musicale.***

Le titre de ce projet est en cours de précision, mais je trouve qu'il reflète l'idée d'ensemble de la réalisation.

- Description :

Le but de ce projet est de créer un outil qui permettrait de réunir en une seule plateforme (interface numérique) un instrument modulaire, qui comprendrait de la synthèse ainsi que des séquenceurs et des effets audio numériques.

Il sera également capable de générer des effets sur un échantillonneur vidéo qui affichera de manière synchronisée en tout temps de la vidéo mise en relation avec le son.

Cette fenêtre vidéo fait partie intégrante de cet instrument, je veux ici concevoir un outil qui repose fondamentalement sur les deux aspects de traitement, audio et vidéo, une sorte de synthétiseur interactif (à la vidéo).

Cet outil comportera donc un synthétiseur modulaire, un échantillonneur audio, des traitements en temps réel pour l'audio et la vidéo, une plateforme vide de montage interactif (utilisation similaire à **Résolum Arena 5**).

Je voudrais contrôler l'ensemble des paramètres de l'interface numérique grâce à un contrôleur Midi (Maschine MKII de chez native instrument.)

Il serait également possible de rajouter un contrôle Arduino, doté de capteurs pour donner une dimension plus scénique lors de la performance.

La finalité du projet serait de pouvoir utiliser cet outil pour composer une performance live et permettre par la suite de devenir un outil de performance selon les différentes compositions possibles.

Sachant que la synthèse est modulaire, il suffit de changer le séquenceur et les réglages.

L'échantillonneur permet d'ajouter d'autres sons, et le traitement vidéo live permet d'interchanger les fichiers vidéo à volonté.

C'est le montage et les effets additifs qui sont interactifs au son, donc il est possible avec un outil de cet ensemble de parties modulables de changer d'échantillon très facilement.

Je vais également essayer de pousser dans cette direction le développement, afin que l'outil soit facile d'accès avec un design simple pour que d'autres musiciens puissent projeter de la vidéo interactive pendant leur performance live.

## Analyse des besoins :

Je vais utiliser **Pure Data** pour réaliser ce projet, car cette plateforme, déjà modulaire dans sa pensée de programmation graphique, me permettrait de réaliser tous les différents modules audio sans beaucoup de difficulté.

Concernant la partie vidéo, la manière la plus simple de traitement de ce module-là serait avec la librairie **GEM**.

Pour d'autres besoins, comme l'addition de certains outils d'acquisition de données Midi, je suis en train d'évaluer encore mes besoins, mais sachant que cette partie est la fin de la conception d'architecture, je pense me pencher sur cette question plus tard.

Concernant le contrôle sur la plateforme Arduino, j'ai déjà les deux librairies qui conviennent à cette tâche : **Comport** et **Mr.Peach**.

Pour la conception de l'architecture de la patch sur **pure Data**, je pense créer plusieurs objets PD (les modules principaux), qui comporteront eux-mêmes des objets PD (secondaires, servant la programmation de chaque module).

Pour se donner une idée des différents modules principaux suivis de leurs modules internes secondaires :

### **Synthétiseur modulaire :**

- OSC 1 : (frequency, fine, tune, square, sine, triangle, saw, sub) frequency modulation?
- OSC 2: (frequency, fine, tune, square, sine, triangle, saw, sub) frequency modulation?
- OSC 3 : (frequency, fine, tune, square, sine, triangle, saw, sub) normal
- Modulation: (triangle, square, sine) même chose qu'un LFO
- Filtre : (cut off frequency, resonance, HP, BP, LP, receives signal for mod)
- Mixer: (vol channel 1, vol channel 2, vol channel 3)
- ADSR : (attack, decay, sustain, release, receive signal for mod)
- AMP : (vol général)
- Signal out: (1, 2, 3)

### **Séquencer 16 pas pour la synthèse de 3 pistes :**

- Tempo : BPM
- Transport : play, stop
- Grille de temps (step size) : 1/4, 1/8, 1/16, 1/32
- Start step : 1-16 (possibilité d'afficher les notes sur chaque pas, mais également un contrôle de position classique.)

Je n'ai pas encore une idée précise de ce qui est faisable pour l'interface graphique, mais j'ai relevé plusieurs travaux effectués par la communauté dont je pourrais m'inspirer.

### **Échantillonneur et séquenceur audio 32 pas/32 pistes :**

- Capacité a enregistré un échantillon en live à l'aide d'un micro.
- Échantillonneur ou l'on peut placer ses objets audio de 32 entrées (a pour but de gérer la rythmique).

Je suis encore en train de réfléchir à l'architecture de ce module où je risque de rencontrer quelques difficultés.

Il est complexe, car je dois pouvoir permettre la lecture et l'écriture d'un fichier audio, et le router vers le séquenceur en temps réel.

### **Module de traitement audio applicable à chaque son via une matrice :**

(Je vais m'inspirer des « *Patch Bay* », pour pouvoir router n'importe quelle piste vers les effets avec un commutateur.)

- Reverb:
- Delay:

## **Module de traitement de la vidéo :**

- Une fenêtre qui affiche la vidéo.
- Un échantillonneur pour les vidéos avec 8 pistes
- Une fenêtre de contrôle
- Paramètres de contrôle avec différents patrons de montage de vidéo réagissant avec le son.
- Différents contrôles d'effet de modulation du signal vidéo
- Différents effets additifs sur la vidéo (Décalage RGB, Postérisation, Bit crusher, Glitch, Miroir H/V).

## **Interface utilisateur (pure data) :**

Je suis en train de concevoir l'architecture à l'aide d'un schéma pour me donner une idée de ce que peut rendre l'interface numérique au niveau des contrôles.

Je préfère commencer à structurer l'interface maintenant pour m'éviter des chemins de programmation trop longs entre mes objets, même si ce n'est pas une priorité pour le moment.

## **Acquisition de connaissances :**

### **Synthétiseur modulaire :**

- Liste d'objets de PD :  
[http://blazicek.net/list\\_of\\_pure\\_data\\_objects.html](http://blazicek.net/list_of_pure_data_objects.html)
- Exemple de synthétiseur modulaire :  
[http://overdrive.music.free.fr/03patchwork\\_synth.php](http://overdrive.music.free.fr/03patchwork_synth.php)

### **séquenceur 16 pas pour la synthèse de 3 pistes :**

- Documentation pour la création de séquenceurs :  
<http://puredata.info/search?SearchableText=sequencer>  
[http://puredata.info/Members/syntax\\_the\\_nerd/seqwiki/Sequencerwikihome/?searchterm=sequencer](http://puredata.info/Members/syntax_the_nerd/seqwiki/Sequencerwikihome/?searchterm=sequencer)

### **Échantillonneur et séquenceur audio 32 pas/32 pistes d'audio :**

*Suite à venir !*

### **Module de traitement audio applicable à chaque son via une matrice :**

*Suite à venir !*

### **Module de traitement de la vidéo :**

*Suite à venir !*

## **Interface utilisateur (côté Pure Data) :**

*Suite à venir !*

## **Modèle :**

*Schéma à venir !*

## **Méthodes :**

### **1 : Synthétiseur modulaire :**

Je voudrais pouvoir assigner à chacun des objets du synthétiseur un contrôle sur l'interface, en commençant par mettre en place plusieurs types de synthèses pour ce qui est des oscillateurs.

*Suite en cours !*